01-12-06

~~$~~ AP/
2182

Am

OIPE IAP22
JAN 1 0 2006
PATENT & TRADEMARK OFFICE

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

PATENT APPLICATION

ATTORNEY DOCKET NO. 10004560-1

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s):	Tom Milner et al.	Confirmation No.:	4840
Application No.:	09/846,645	Examiner:	J.D. Schneider
Filing Date:	05/01/2001	Group Art Unit:	2182

Title: SYSTEM AND METHOD FOR IDENTIFICATION OF DEVICES ASSOCIATED WITH INPUT/OUTPUT PATHS

Mail Stop Appeal Brief-Patents
Commissioner For Patents
PO Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Transmitted herewith is the Appeal Brief in this application with respect to the Notice of Appeal filed on 11/10/2005.

The fee for filing this Appeal Brief is (37 CFR 1.17(c)) $500.00.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

□(a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d)) for the total number of months checked below:

□ 1st Month $120 □ 2nd Month $450 □ 3rd Month $1020 □ 4th Month $1590

□ The extension fee has already been filed in this application.

☒(b) Applicant believes that no extension of time is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account 08-2025 the sum of $ 500. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees. A duplicate copy of this sheet is enclosed.

Respectfully submitted,

Tom Milner et al.

By

Thomas J. Meaney

Attorney/Agent for Applicant(s)

Reg No. : 41,990

Date : January 10, 2006

Telephone : (214) 855-8230

(X) I hereby certify that this correspondence is being deposited with the U.S. Postal Service as Express Mail, Airbill No. EV482724211US, in an envelope addressed to: MS Appeal Brief – Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Date of Deposit: January 10, 2006

Typed Name: Robert L. Greeson

Signature: Robert Greeson

O I P E
JAN 1 0 2006
IAP22
PATENT & TRADEMARK OFFICE

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

Docket No.: 10004560-1
(PATENT)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Tom Milner et al.

Application No.: 09/846,645

Confirmation No.: 4840

Filed: May 1, 2001

Art Unit: 2182

For: SYSTEM AND METHOD FOR IDENTIFICATION OF DEVICES ASSOCIATED WITH INPUT/OUTPUT PATHS

Examiner: J. D. Schneider

01/13/2006 SHASSEN1 00000007 082025 09846645
01 FC:1402 500.00 DA

APPEAL BRIEF

MS Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

As required under 37 C.F.R. § 41.37(a), this brief is filed within two months of the Notice of Appeal filed in this case on November 10, 2005, and is in furtherance of said Notice of Appeal.

The fees required under 37 C.F.R. § 41.20(b)(2) are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief contains items under the following headings as required by 37 C.F.R. § 41.37 and M.P.E.P. § 1206:

I. Real Party In Interest
II Related Appeals and Interferences
III. Status of Claims
IV. Status of Amendments
V. Summary of Claimed Subject Matter

I. REAL PARTY IN INTEREST

The real party in interest for this appeal is:

Hewlett-Packard Development Company, L.P., a Texas Limited Partnership having its principal place of business in Houston, Texas.

II. RELATED APPEALS, INTERFERENCES, AND JUDICIAL PROCEEDINGS

There are no other appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

III. STATUS OF CLAIMS

A. Total Number of Claims in Application

There are 23 claims pending in application.

B. Current Status of Claims

1. Claims canceled: 1-13

2. Claims overruled from consideration but not canceled: None

3. Claims pending: 14-36

4. Claims allowed: None

5. Claims rejected: 14-36

C. Claims On Appeal

The claims on appeal are claims 14-36

IV. STATUS OF AMENDMENTS

Applicant did not file an Amendment after the Final Rejection, which was mailed on September 22, 2005 (hereinafter the "Final Action").

V. SUMMARY OF CLAIMED SUBJECT MATTER

Claim 14 recites a method for discovering a type of device associated with an input/output (I/O) path of a storage area network. The method includes retrieving a plurality of property files from a predefined subdirectory, wherein each property file of said plurality of property files describes a type of device (pg. 10, lines 10-15). The method also includes removing a class identifier from each property file of said property files, wherein each class identifier identifies a class (pg. 10, lines 20-25). The method further includes creating an object of the respective class of each class identifier (pg. 10, lines 24-26), and calling a specific method, from a plurality of methods, for each created object, wherein said specific method is operable to determine whether a device associated with said I/O path is the type of device described by the property file associated with said object (pg. 10 line 24- pg. 11 line 5).

Claim 15 recites a method for discovering a type of device associated with an input/output (I/O) path of a storage area network further comprises. The method includes adding a new storage device to said storage area network. The method also includes wherein said new storage device is caused to be associated with said I/O path, and wherein said new storage device is a new type of device to said storage area network (pg. 7, lines 8-10).

Claim 17 recites a method for discovering a type of device associated with an input/output (I/O) path of a storage area network wherein a default property file of said plurality of property files identifies a simple network management protocol (SNMP) class, wherein said default SNMP class defines a method to identify devices by a comparing a SNMP system object identifier to at least one field in said default property file (pg. 8, lines 18-27).

Claim 18 recites a system for analyzing input/output (I/O) paths of a storage area network (SAN) comprises: a plurality of servers, wherein said servers are communicatively coupled to a fabric of said SAN (pg. 6, lines 5-10). The system includes a plurality of host agent processes, wherein each of said host agent processes executes on a respective server of said plurality of servers, and wherein said host agent processes are operable to query devices associated with host logical unit numbers I/O paths of said SAN to gather device information (pg. 7, lines 18-27). The system also includes a management server, wherein said management server employs a simple network management protocol (SNMP) manager process to query devices associated with SNMP I/O paths of said SAN to gather device information (pg. 7, lines 11-17). The system further includes a plurality of property files stored in a predefined directory, wherein each property file of said plurality of property files describes a type of device, and wherein each property file of said plurality of property files includes an identifier of code operable to determine whether a device associated with an I/O path is the type of device described by its associated property file (pg. 10, lines 10-15). The system further includes a management server process, wherein said management server process is operable to receive gathered device information from said plurality of host agent processes and from said SNMP manager process. Finally, the system includes wherein said management server process is operable to call code identified by property files with gathered device information as arguments to thereby uniquely identify the devices associated with said I/O paths of said SAN (pg. 7 line 11- pg. 8 line 9)

Claim 21 recites a method for identifying a device associated with an input/output (I/O) path comprises: retrieving device information from a target device associated with said I/O path utilizing a device control protocol (pg. 8, lines 19-26; pg. 10 line 27- pg. 11 line 5). The method includes retrieving a property file defining a device, wherein said property file designates a code set for identifying, from a plurality of different code sets for identifying (pg. l0, lines 10-15). The method also includes executing said designated code, wherein said designated code set utilizes said retrieved information to determine whether said target device is said device defined by said property file (pg. 10 line 18- pg 12 line 27)

According to claim 26, the method for determining the nature of a device associated with an input/output (I/O) path comprises: retrieving device information from a target device associated with said I/O path utilizing a device control protocol (pg. 10 line 27- pg. 11 line 5), retrieving a property defining the nature of a known device (pg. 10 line 27- pg. 11 line 20), and executing code associated with said property file, wherein said code is operable to uniquely identify said target device, and operable to determine whether or not said property file defines the nature of said uniquely identified device (pg. 10 line 18- pg 12 line 27).

According to claim 31, the system for determining the nature of a target device associated with an input/output (I/O) path comprises: at least two device identifying code sets, wherein each said code set is operable to identify a different group of devices (pg 10, line 4- pg. 12, line 27), at least two property files, wherein each said property file defines a different device type, and wherein each said property file is associated with a different identifying code set (pg 10, line 4- pg. 12, line 27), and a processor operable to call one said property file and execute said identifying code set associated with said called property file, wherein said identifying code set associated with said called property file determines if said target device is a member of the group defined by said called property file (pg 10, line 4- pg. 13, line 10).

VI. GROUNDS OF OBJECTION TO BE REVIEWED ON APPEAL

A. Whether claims 21-36 are properly rejected under 35 U.S.C 112, first paragraph.

B. Whether claims 14, 15, and 17 are properly rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,122,639 to Babu et al (hereinafter "Babu").

C. Whether claims 16 and 18 are properly rejected under 35 U.S.C. 103(a) as being unpatentable over Babu in view of U.S. Patent Application Publication No. 2002/0161852 to Allen et al (hereinafter "Allen").

D. Whether claims 19 and 20 are properly rejected under 35 U.S.C. 103(a) as being unpatentable over Babu and "the applicant admitted prior art (AAPA)" in further view of Allen.

VII. ARGUMENT

<u>Rejection Under 35 U.S.C. § 112, first paragraph</u>

The Final Action rejects claims 21-36 for failing to comply with the enablement requirement of 35 U.S.C. § 112, first paragraph. Specifically, the Examiner states that the claims contain subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the invention. The Applicant respectfully submits, however, that the specification as originally filed, enables claims 21-36.

With respect to claims 21, 26, and 31 the Examiner contends that the specification does not teach that "the code retrieved from the device for identifying the type of device is executable code." (*see* Final Action, paragraph 7). However the description of a software element's function (i.e., identifying the type of device) is considered adequate for enablement by the Federal Circuit and under M.P.E.P. § 2106.01, because one of ordinary skill in the art is capable of writing code to fulfill that function. "As a general rule, where software constitutes part of a best mode of carrying out an invention, description of such a best mode is satisfied by a disclosure of the functions of the software . . . [t]his is because, normally, writing code for such software is within the skill of the art." *Fonar Corp. v. General Electric Co.*, 107 F.3d 1543 (Fed. Cir. 1997). The software function with regard to claims 21, 26, and 31 is sufficiently detailed in the specification to enable its use by one of ordinary skill in the art. (*see* pgs. 7-10). Therefore, the Appellant respectfully requests that the 35 U.S.C. § 112 rejection of claims 21, 26, and 31 be overruled.

With respect to claims 22 and 30, the Examiner states that "the specification never refers to a SysObjID. Applicant claims using a system object identifier in claim 30." (*see* Final Action, paragraph 8). However, the Appellant respectfully points out that a system object ID (or SysObjID) is explicitly referred to, at least at pg. 8 line 26 and pg. 10 line 8, of the patent application as filed.

Finally, claims 23-25, 27-30, and 32-36 are rejected for including the non-enabled subject matter of the claims from which they depend. In view of the remarks above, that is, Applicant's arguments regarding the enablement of claims 21, 22, 26, 30, and 31, Applicant submits each of claims 23-25, 27-30, and 32-36 are also enabled.

Rejection Under 35 U.S.C. § 103(a); Babu

Claims 14, 15, and 17 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Babu.

To establish a prima facie case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art cited must teach or suggest all the claim limitations. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed Cir. 1991). Without conceding the second criteria has been met, the Appellant respectfully asserts that the Final Action fails to establish a prima facie case of obviousness because the Examiner's proposed combination fails to teach or suggest all of the Appellant' claimed elements, and fails to demonstrate a motivation for combining the cited references.

Claim 14 recites,

> "calling a specific method, from a plurality of methods, for each created object, wherein said specific method is operable to determine whether a device associated with said I/O path is the type of device described by the property file associated with said object."

In the Final Action, the Examiner points to Babu, at col. 2 line 64- col. 3 line 10, to satisfy this element. (Final Action, paragraph 13). However, Babu merely describes "obtaining a device type identifier from the device." (Babu col. 2, lines 64-65). Babu does "[look] up the device type identifier in a device type table stored in the database." (Babu col. 3, lines 8-9). However, this step does not determine whether the device is the type of device described by the property file associated with the object method, as the Examiner contends. When Babu performs this step, the device type has already been determined. In an attempt to cure this defect, the Examiner contends that elements 302-314 of Babu "retriev[e] property files from a directory where the property files describes a type of device." The Appellant respectfully points out, however, that even if true, such a showing would not meet the elements of the claim. Claim 14 is a method "for discovering a type of device associated with an input/output (I/O) path of a storage area network." Elements 302-314 in Babu could not be used for discovering the type of device because the device type is the very information these steps of Babu use to obtain the device type identifier. Babu does not, therefore, teach or suggest every element of claim 14. The Appellant asserts that the Examiner has failed to establish a prima facie case of obviousness and respectfully request that the Appeal Board overrule the rejection of claim 14.

Claim 15 recites,

> "adding a new storage device to said storage area network, wherein said new storage device is caused to be associated with said I/O path, and wherein said new storage device is a new type of device to said storage area network."

In the Final Action, the Examiner points to Babu, at col. 1 lines 44-55 and col. 2 line 64- col. 3 line 67, to satisfy this limitation. (Final Action, paragraph 14). However, these selected portions of Babu merely describe a network information collection mechanism that can adapt to new devices. (*see* Babu col. 1, lines 44-46). Clearly, Babu does not teach or suggest a new storage device wherein the new storage device is associated with the I/O path, and wherein the new storage device is a new type of device to the storage area network as recited in claim 15. As shown, Babu fails to teach or suggest each element of the Appellant's claimed invention. Therefore, the Appellant asserts that the Examiner has failed to establish a prima facie case of obviousness and respectfully requests that the Appeal Board overrule the rejection of claim 15.

Claim 17 recites,

> "wherein a default property file of said plurality of property files identifies a simple network management protocol (SNMP) class, wherein said default SNMP class defines a method to identify devices by a comparing a SNMP system object identifier to at least one field in said default property file."

In the Final Action, the Examiner points to Babu, at Fig. 5, col. 8. lines 7-24 and col. 2 lines 65- col. 3 lines 54, to satisfy this limitation. (*see* Final Action, paragraph 15). The Appellant respectfully submits that Babu, at Fig. 5, is wholly silent as to the recited limitation. Further, at these selected portions of Babu it merely discloses querying a database having a Device Type table, where rows of the table correspond to a SysObjID value. (*see* Babu at col. 8 lines 13-20). The Appellant respectfully asserts that querying a database table, as described in Babu, is not the same as identifying a simple network management protocol (SNMP) class as recited in claim 17. Moreover, the lookup operation of Babu does not compare a system object identifier to a property file. Rather, in Babu, the SysObjID is a table component merely used as a key to look up descriptive information also contained in the table. (*see id*). As shown, Babu fails to teach or suggest each element of the Appellant's claimed invention. Therefore, the Appellant asserts that the Examiner has failed to establish a prima facie case of obviousness and respectfully requests that the Appeal Board overrule the rejection of claim 17.

Rejection Under 35 U.S.C. § 103(a); Babu in View of Allen

Claims 16 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Babu in view Allen.

To establish a prima facie case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art cited must teach or suggest all the claim limitations. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed Cir. 1991). Without conceding the second criteria has been met, the Appellant respectfully asserts that the Final Action fails to establish a prima facie case of obviousness because the Examiner's proposed combination fails to teach or suggest all of the Appellant' claimed elements, and fails to demonstrate a motivation for combining the cited references.

Claims 16 depends from claim 14 and inherits each limitation therefrom. As stated above, Babu does not teach or suggest determining whether the device is the type of device described by the property file associated with the object method. Further, Allen does not appear to teach or suggest the missing elements nor does the Examiner rely on Allen to teach or suggest such elements. Allen merely teaches fibre channel tracking in unknown configurations. Therefore, the Appellant respectfully requests that the Appeal Board overrule the rejection of claim 16.

Claim 18 recites a system for analyzing input/output paths that includes:

> a management server process, wherein said management server process is operable to receive gathered device information from said plurality of host agent processes and from said SNMP manager process; and wherein said management server process is operable to call code identified by property files with gathered device information as arguments to thereby identify types of devices associated with I/O paths of said SAN.

In the Final Action, the Examiner points to Babu, at col. 3 lines 46-67 and col. 14 lines 62-col. 15 line 6, to satisfy this limitation. (*see* Final Action, paragraph 19). Babu sends "an SNMP Query For a system object identifier to the network . . . and [tests] whether the device is discovered in the network." (*see* Babu col. 3, lines 47-49). However, this step does not identify types of devices, as the Examiner contends, because when this step is performed in Babu, the device type has already been determined. Moreover, it would be illogical to interpret this "test" step as identifying types of devices, because the type of device "appears to be the very information of Babu uses to perform the "test." Allen does not teach or suggest this element either, and indeed, the Examiner does not rely on it to do so. The combination of Babu and Allen, therefore, does not teach or suggest every element of claim 18. As such, the Appellant asserts that the Examiner has failed to establish a prima facie case of obviousness and respectfully requests that the Appeal Board overrule the rejection of claim 18.

Claim 18 also recites:

> a plurality of property files stored in a predefined directory, wherein each property file of said plurality of property files describes a type of device, and wherein each property file of said plurality of property files includes an identifier of code operable to determine whether a device associated with an I/O path is the type of device described by its associated property file

The Appellant respectfully asserts that the proposed combination of Babu and Allen fails to teach this limitation. In trying to meet this limitation, the Examiner attempts to equate "property files" with the element 310 of Babu. However, the element 310 is not a "property files" because, among other things: a) element 310 comes from the device itself, not from a predefined directory; and b) element 310 does not appear to contain "code operable to determine whether a device associated with an I/O path." Although not relied upon to do so, Allen does not teach this limitation either. Thus, the combination of Babu and Allen does not teach or suggest all of the limitations of claim 18. Therefore, the Examiner has failed to establish a prima facie case of obviousness. The Appellant respectfully asks the Appeal Board to overrule the rejection to claim 18.

Rejection Under 35 U.S.C. § 103(a); Babu and (AAPA) in View of Allen

Claims 19 and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Babu and (AAPA) in further view of Allen.

Claims 19 and 20 depend directly or indirectly from claim 18, and thus inherit all of the limitations of claim 18. As shown, the combination of Babu and Allen does not teach or suggest a management server process, wherein said management server process is operable to receive gathered device information from said plurality of host agent processes and from said SNMP manager process. Therefore, claims 19 and 20, through their dependence on claim 18, contain limitations not taught or suggested by Babu or Allen. The Appellant respectfully submits that claims 19 and 20 are patentable over Babu and Allen. With regard to the Examiner's statements regarding AAPA, it is unclear from the Final Action, or any previous Action, what the Examiner views as "applicant admitted prior art," or (AAPA) Moreover, the Examiner makes no reference as to what the AAPA is relied upon to teach or suggest. Therefore, the Appellant respectfully submits that the Examiner's rejection fails to comport with 35 U.S.C. 103(a). The Appellant respectfully request that the Appeal Board overrule the rejection. Further, the Appellant is unable to determine

VIII. CLAIMS

A copy of the claims involved in the present appeal is attached hereto as Appendix A.

IX. EVIDENCE

No evidence pursuant to 37 C.F.R. §§ 1.130, 1.131, or 1.132 or entered by or relied upon by the examiner is being submitted.

X. RELATED PROCEEDINGS

No related proceedings are referenced in II. above, or copies of decisions in related proceedings are not provided, hence no Appendix is included.

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as Express Mail, Airbill No. EV482724211US in an envelope addressed to: MS Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Date of Deposit: January 10, 2006

Typed Name: Robert L. Greeson

Signature: Robert Greeson

Respectfully submitted,

By____________________
Thomas J. Meaney
Reg. No.: 41,990
Date: January 10, 2006
Telephone No. (214) 855-8230

APPENDIX A

Claims Involved in the Appeal of Application Serial No. 09/846,645

1-13 (Canceled)

14. (Previously Presented) A method for discovering a type of device associated with an input/output (I/O) path of a storage area network, comprising:

(a) retrieving a plurality of property files from a predefined subdirectory, wherein each property file of said plurality of property files describes a type of device;

(b) removing a class identifier from each property file of said property files, wherein each class identifier identifies a class;

(c) creating an object of the respective class of each class identifier; and

(d) calling a specific method, from a plurality of methods, for each created object, wherein said specific method is operable to determine whether a device associated with said I/O path is the type of device described by the property file associated with said object.

15. (Original) The method of claim 14 further comprising:

(e) adding a new storage device to said storage area network, wherein said new storage device is caused to be associated with said I/O path, and wherein said new storage device is a new type of device to said storage area network;

(f) storing a new property file in said predefined subdirectory describing said new type of device; and

(g) restarting code of a management server to thereby cause repetition of steps (a)-(d) utilizing said new property file.

16. (Original) The method of claim 14 wherein a default property file of said plurality of property files identifies a default small computer system interface (SCSI) class, wherein said default SCSI class defines a method to identify devices by comparing SCSI vendor identifier and product identifier information to at least one field in said default property file.

17. (Original) The method of claim 14 wherein a default property file of said plurality of property files identifies a simple network management protocol (SNMP) class, wherein said default SNMP class defines a method to identify devices by a comparing a SNMP system object identifier to at least one field in said default property file.

18. (Previously Presented) A system for analyzing input/output (I/O) paths of a storage area network (SAN) comprising:

a plurality of servers, wherein said servers are communicatively coupled to a fabric of said SAN;

a plurality of host agent processes, wherein each of said host agent processes executes on a respective server of said plurality of servers, and wherein said host agent processes are operable to query devices associated with host logical unit numbers I/O paths of said SAN to gather device information;

a management server, wherein said management server employs a simple network management protocol (SNMP) manager process to query devices associated with SNMP I/O paths of said SAN to gather device information;

a plurality of property files stored in a predefined directory, wherein each property file of said plurality of property files describes a type of device, and wherein each property file of said plurality of property files includes an identifier of code operable to determine whether a device associated with an I/O path is the type of device described by its associated property file; and

a management server process, wherein said management server process is operable to receive gathered device information from said plurality of host agent processes and from said SNMP manager process; and wherein said management server process is operable to call code identified by property files with gathered device information as arguments to thereby uniquely identify the devices associated with said I/O paths of said SAN.

19. (Original) The system of claim 18 wherein said management server process, includes:

code for creating an array of identifiers including each said identifier from each property file;

code for instantiating a plurality of small computer system interface (SCSI) device discovery objects utilizing identifiers from said array that identify SCSI device classes; and

code for instantiating a plurality of SNMP device discovery objects utilizing identifiers from said array that identify SNMP device classes.

20. (Original) The system of claim 19 wherein said management server process includes:

code for calling a method of each instantiated SCSI device discovery object for each host logical unit numbers I/O path; and

code for calling a method of each instantiated SNMP device discovery object for each SNMP I/O path.

21. (Previously Presented) A method for identifying a device associated with an input/output (I/O) path, comprising:

retrieving device information from a target device associated with said I/O path utilizing a device control protocol;

retrieving a property file defining a device, wherein said property file designates a code set for identifying, from a plurality of different code sets for identifying;

executing said designated code, wherein said designated code set utilizes said retrieved information to determine whether said target device is said device defined by said property file.

22. (Previously Presented) The method of claim 21 wherein said device information is not a SysObjID.

23. (Previously Presented) The method of claim 21 wherein said property file identifies a class defining said type of device, said method further comprising:

instantiating an object of said class;

wherein said step of executing code includes calling a method of said instantiated object.

24. (Previously Presented) The method of claim 23 wherein said executing code determines that said target device is said type of device, said method further comprising:

calling a second method of said instantiated object to create a unique identifier for said device.

25. (Previously Presented) The method of claim 21 wherein said code is operable to query said target device for additional device information.

26. (Previously Presented) A method for determining the nature of a device associated with an input/output (I/O) path, said method comprising:

retrieving device information from a target device associated with said I/O path utilizing a device control protocol;

retrieving a property defining the nature of a known device;

executing code associated with said property file, wherein said code is operable to uniquely identify said target device, and operable to determine whether or not said property file defines the nature of said uniquely identified device.

27. (Previously Presented) The method of claim 26 wherein said executed code further determines the device type of said target device.

28. (Previously Presented) The method of claim 26 wherein the unique identity of said target device is capable of being determined in a plurality of device control protocols.

29. (Previously Presented) The method of claim 26 wherein said target device is a small computer system interface (SCSI) device, and wherein said step of retrieving said device information includes obtaining a vendor identifier and a product identifier of said target device from a host agent.

30. (Previously Presented) The method of claim 26 wherein said target device is an simple network management protocol (SNMP) device, and wherein said step of retrieving includes obtaining a SNMP system object identifier of said target device.

31. (Previously Presented) A system for determining the nature of a target device associated with an input/output (I/O) path, said system comprising:

at least two device identifying code sets, wherein each said code set is operable to identify a different group of devices;

at least two property files, wherein each said property file defines a different device type, and wherein each said property file is associated with a different identifying code set; and

a processor operable to call one said property file and execute said identifying code set associated with said called property file, wherein said identifying code set associated with said called property file determines if said target device is a member of the group defined by said called property file.

32. (Previously Presented) The system of claim 31 wherein each said identifying code set is capable of uniquely identifying a device.

33. (Previously Presented) The system of claim 31 wherein the nature of said target device can be determined in a plurality of device control protocols.

34. (Previously Presented) The system of claim 33 wherein said target device can be uniquely identified regardless of the device control protocol.

35. (Previously Presented) The system of claim 31 wherein at least one said identifying code set is operable to communicate with a host agent to obtain information utilized to determine whether said target device is the type of device defined by the property file associated with said identifying code set.

36. (Previously Presented) The system of claim 35 wherein said host agent provides an application programming interface (API) to obtain said information.